

**In the Claims:**

Please amend claims 2-4, 8, 12, 14, 18, and 21-28 as indicated below.

1. (Original) A system, comprising:

an application server comprising:

an application component whose data is represented in a system default locale;

localization logic configured to translate input parameters or return values between the system default locale and another locale; and

interception logic configured to intercept a plurality of method calls to the application component;

wherein the interception logic is configured to invoke the localization logic and the application component in response to one or more of the plurality of method calls.

2. (Currently amended) The system as recited in claim 1, wherein the localization logic is separate from the application component ~~logic~~.

3. (Currently amended) The system as recited in claim 1, wherein the application component ~~logic~~ is configured to operate on data stored in a primary database table in which the data is represented in the system default locale.

4. (Currently amended) The system as recited in claim 3, wherein the application component ~~logic~~ is configured to invoke one or more Java Data Base Connectivity methods to maintain the primary database table.

5. (Original) The system as recited in claim 1, wherein the localization logic is configured to operate on data stored in one or more localization database tables in which the data is represented in a locale other than the system default locale.

6. (Original) The system as recited in claim 5, wherein the localization logic is configured to invoke one or more JDBC methods to maintain the one or more localization database tables.

7. (Original) The system as recited in claim 1, wherein the interception logic includes one or more dynamic proxies that are configured to intercept application component method calls before the execution of the method.

8. (Currently amended) The system as recited in claim 1, wherein the localization logic is configured to make one or more ~~Java~~ reflection calls to determine whether input parameters to, or return values from the application component are localizable.

9. (Original) The system as recited in claim 1, wherein the localization logic is configured to invoke localization functions before and after the execution of the method.

10. (Original) The system as recited in claim 1, further comprising a metadata file indicating the system default locale and translatable table information for the application component.

11. (Original) A method, comprising:

creating a dynamic proxy for an application component comprising one or more localizable parameters or return values;

modifying a service locator to return an interface to the proxy;

storing one or more localization tables wherein each localization table stores localizable parameters or return values for a locale; and

the dynamic proxy invoking localization logic, separate from application logic, to translate localizable parameters or return values between a system default locale to another locale.

12. (Currently amended) A method, comprising:

intercepting a method call to an application component before or after the execution of the method;

determining whether input parameters or return values associated with the method call are localizable;

localizing input parameters or return values based on data stored in one or more localization tables;

invoking the application component to execute the method;

maintaining a primary data table for the application component; and

maintaining the localization tables.

13. (Original) The method as recited in claim 12, wherein said intercepting comprises calling a dynamic proxy for the application component.

14. (Currently amended) The method as recited in claim 12, wherein said determining comprises making one or more ~~Java~~ reflection calls to the application component.

15. (Original) The method as recited in claim 12, wherein said maintaining of the primary and localization tables comprises invoking one or more JDBC methods to access one or more databases.

16. (Original) The method as recited in claim 12, wherein said maintaining of the primary data table is performed by application component logic.

17. (Original) The method as recited in claim 16, wherein said maintaining of the localization tables is performed by localization logic separate from the application component logic.

18. (Currently amended) The method as recited in claim 12, further comprising accessing from a metadata file information about ~~the~~ a system default locale and translatable table information.

19. (Original) The method as recited in claim 12, wherein said intercepting is performed by logic that is separate from logic for executing the method.

20. (Currently amended) A computer-~~accessible~~-readable storage medium, comprising program instructions, wherein the program instructions are executable to:

create a dynamic proxy for an application component comprising localizable parameters or return values;

modify a service locator to return an interface to the proxy;

store one or more localization tables wherein each localization table stores localizable parameters or return values for a locale; and

implement localization logic, separate from application logic, invoked by the dynamic proxy to translate localizable parameters or return values between a system default locale to another locale.

21. (Currently amended) A computer-~~accessible~~-readable storage medium, comprising program instructions, wherein the program instructions are executable to:

intercept a method call to an application component before or after execution of the method;

determine whether input parameters or return values associated with the method call are localizable;

localize input parameters or return values based on data stored in one or more localization tables;

invoke the application component to execute the method;

maintain a primary data table for the application component; and

maintain the localization tables.

22. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein the program instructions are further executable to call dynamic proxies to intercept the method call to the application component before and after execution of the method.

23. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein the program instructions are further executable to make ~~Java~~ reflection calls to determine whether input parameters or return values associated with the method call are localizable.

24. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein the program instructions are further executable to invoke one or more JDBC methods to maintain primary and localization tables.

25. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein the program instructions of the application component ~~logic~~ are further executable to maintain the primary data table.

26. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein the program instructions of ~~the~~ localization logic are further executable to maintain the localization data tables.

27. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein information about ~~the~~ a system default locale and translatable table information is stored in an XML metadata file.

28. (Currently amended) The computer-~~accessible~~-readable storage medium as recited in claim 21, wherein said intercepting is performed by logic that is separate from logic for executing the method.